

Amendments to the Claims

1. (Currently Amended) A method of handling events received at
2 sockets in a computer server configured to serve multiple clients, the method comprising:
executing a polling module configured to poll server sockets to detect events
4 received at said sockets;
registering a first plurality of sockets with said polling module for serving the
6 multiple clients, wherein each of said sockets in said first plurality of sockets is
associated with an event consumer and the plurality of sockets includes:
8 a single data delivery socket configured for simultaneously serving media
data to all of the multiple clients; and
10 for each client, a separate control socket configured to receive command
events from the client;
12 notifying a first event consumer associated with a first socket in said first plurality
of sockets when a first event is received at said first socket; and
14 invoking a task configured to facilitate handling of said first event;
wherein a first processor thread is shared among said first plurality of sockets for
16 said polling; and
wherein one or more processor threads are allocated to the execution of tasks
18 invoked by said event consumers.

2. (Original) The method of claim 1, further comprising:
2 registering a second plurality of sockets with said polling module, wherein each
of said sockets in said second plurality of sockets is associated with an event consumer;
4 wherein a second processor thread is shared among said second plurality of
sockets.

3. Cancelled

4. (Currently Amended) The method of claim 1 3, wherein said event
2 consumers are program objects, and each of said event consumers is one of the set of:

a listener consumer configured to handle a connection request event;
4 a connection consumer configured to handle a media streaming command event;
and
6 a receiver consumer configured to handle a media stream quality event;
wherein said listener consumer, said connection consumer and said receiver
8 consumer are derived from a single abstract base class.

5. (Currently Amended) The method of claim 1 3, wherein said first
2 event comprises a connection request from a client; and
wherein said first event consumer is a listener event consumer configured to
4 establish a client connection through a second socket in response to said connection
request.

6. (Currently Amended) The method of claim 1 3, wherein said first
2 event comprises a media streaming command; and
wherein said first event consumer is a connection consumer configured to execute
4 said media streaming command.

7. (Currently Amended) The method of claim 1 3, wherein said first
2 event comprises media stream quality information; and
wherein said first event consumer is a receiver consumer configured to adjust said
4 media stream according to said media stream quality information.

8. (Currently Amended) The method of claim 1 3, wherein said first
2 socket is configured to host connections with multiple clients simultaneously.

9. (Currently Amended) The method of claim 1 3, wherein said first
2 event comprises a request for a media streaming control connection from a first client and
said first event consumer is a listener event consumer, and wherein said invoking a task
4 comprises:
establishing a media streaming control connection with the first client through a

6 second socket configured for media streaming control connections with multiple clients;
wherein said second socket is associated with a connection event consumer
8 configured to handle a media streaming control command.

10. (Currently Amended) The method of claim 9, further comprising:
2 receiving, from the first client at said second socket, a media streaming command
to stream media to the first client; and
4 establishing a media streaming quality connection with the first client through a
third socket configured for media streaming quality connections with multiple clients;
6 wherein said third socket is associated with a receiver event consumer configured
to handle media streaming quality information.

11. (Original) A method of handling events received at a server
2 configured to stream media to clients, wherein processor resources within the server are
allocated in the form of threads, comprising:
4 polling one or more registered server sockets for events received at the server
from clients, wherein each registered socket is associated with an event consumer
6 configured to handle an event received at the registered socket;
receiving a client connection request at a first socket;
8 notifying a first event consumer of the connection request, wherein said first event
consumer is associated with said first socket;
10 registering a second socket configured to receive media streaming commands
through a connection established in response to said client connection request;
12 receiving at said second socket a media streaming command from the client;
notifying a second event consumer of the command, wherein said second event
14 consumer is associated with said second socket; and
issuing one or more tasks configured to execute the media streaming command;
16 wherein a first thread is shared among a first collection of sockets comprising said
first socket and said second socket; and
18 wherein a set of threads is allocated to said one or more tasks.

12. (Original) The method of claim 11, further comprising:
2 registering a third socket configured to receive data concerning the quality of
media being streamed to the client, wherein said third socket is associated with a third
4 event consumer;

wherein said first collection of sockets includes said third socket.

13. (Original) The method of claim 11, wherein said second socket is
2 configured to receive media streaming commands from multiple different clients.

14. (Original) A computer readable storage medium storing instructions
2 that, when executed by a computer, cause the computer to perform a method of handling
events received at a server configured to stream media to clients, wherein processor
4 resources within the server are allocated in the form of threads, the method comprising:
polling one or more registered server sockets for events received at the server
6 from clients, wherein each registered socket is associated with an event consumer
configured to handle an event received at the registered socket;
8 receiving a client connection request at a first socket;
notifying a first event consumer of the connection request, wherein said first event
10 consumer is associated with said first socket;
registering a second socket configured to receive media streaming commands
12 through a connection established in response to said client connection request;
receiving at said second socket a media streaming command from the client;
14 notifying a second event consumer of the command, wherein said second event
consumer is associated with said second socket; and
16 issuing one or more tasks configured to execute the media streaming command;
wherein a first thread is shared among a first collection of sockets comprising said
18 first socket and said second socket; and
wherein a set of threads is allocated to said one or more tasks.

15. Cancelled

16. Cancelled

17. Cancelled

18. Cancelled

19. Cancelled

20. Cancelled

21. Cancelled

22. Cancelled

23. Cancelled

24. (New) A media streaming server configured to stream media to
2 multiple clients, the media streaming server comprising:
a set of server sockets, including:
4 for each of the multiple clients, a separate control socket configured to
receive media stream command events from the client;
6 a single delivery socket configured to stream the media to the multiple
clients simultaneously; and
8 a single quality control socket configured to receive media quality
information events from the multiple clients;
10 a poller module configured to:
poll one or more sockets in the set of sockets; and
12 when an event is received at a polled socket, pass the event to a consumer
object configured to handle the event;
14 a set of consumer object classes derived from a single abstract base class of

consumer objects, wherein the set of consumer object classes includes:

16 a listener object class configured to listen for connection requests from
new clients;

18 a connection object class configured to handle command events received
at polled sockets coupled to the multiple clients; and

20 a receiver object class configured to process data received at a polled
socket; and

22 a set of task objects derived from a single abstract base class of task objects,
wherein the set of task objects includes:

24 a command task object configured to execute a command received as part
of a command event; and

26 a receive data task object configured to read data received at the media
streaming server.

25. (New) The media streaming server of claim 24, further
2 comprising:

 one or more poll tables configured to identify said control sockets, said delivery
4 socket and said quality control socket for polling.

26. (New) The media streaming server of claim 24, wherein a given
2 consumer object derived from one of said consumer object classes is configured to:
 register a socket with said poller when the consumer object is responsible for
4 handling an event received at the socket.

27. (New) The media streaming server of claim 26, wherein the given
2 consumer object is further configured to:
 place a task object in a task queue to handle an event received at the registered
4 socket.

28. (New) The media streaming server of claim 27, further
2 comprising:

4 a first set of processor threads dedicated to executing task objects placed in the
task queue.

29. (New) The media streaming server of claim 28, further
2 comprising:
a second set of processor threads dedicated to polling sockets registered with said
4 poller.

30. (New) The media streaming server of claim 24, wherein the media
2 streaming server:
uses RTSP (Real-Time Streaming Protocol) to receive media stream command
4 events via the control sockets;
uses RTP (Real-time Transfer Protocol) to stream the media via the delivery
6 socket; and
uses RTCP (Real-time Transport Control Protocol) to receive media quality
8 information events via the quality control socket.

31. (New) A method of handling an event received at a media server
2 from one of multiple clients receiving streamed media from the media server, the method
comprising:
4 (a) polling server sockets identified in one or more polltables, wherein one or
more processor threads are dedicated to each polltable for polling sockets identified in the
6 polltable;
(b) executing a first listener object configured to listen at a listener socket for
8 connections from the clients;
(c) adding the listener socket to one of the polltables;
10 (d) in response to receipt of a first client connection at the listener socket:
(1) waking a processor thread dedicated to the polltable that includes
12 the listener socket;
(2) identifying the first listener object as the object responsible for
14 handling events received at the listener socket; and

16 (3) creating a first control socket dedicated to receiving control
 commands from the first client connection;
 (e) executing a first connection object configured to accept control commands
 18 from client connections;
 (f) adding the first control socket to one of the polltables;
 20 (g) in response to receipt of a control command at the first control socket:
 (1) waking a processor thread dedicated to the polltable that includes
 22 the first control socket;
 (2) identifying the first connection object as the object responsible for
 24 handling events received at the first control socket;
 (3) creating a first control task object configured to handle the control
 26 command; and
 (4) executing the first control task object;
 28 (h) streaming the media to the multiple clients simultaneously through a
 single media delivery socket; and
 30 (i) receiving streaming quality information from the multiple clients at a
 single quality control socket.

32. (New) The method of claim 31, wherein (b) comprises:
 2 (1) deriving a listener class of objects from an abstract base class of consumer
 objects; and
 4 (2) instantiating said first listener object from said listener class of objects.

33. (New) The method of claim 31, wherein adding a socket to a
 2 polltable comprises initiating polling of the socket.

34. (New) The method of claim 31, wherein (d) further comprises:
 2 (4) ceasing polling of the listener socket until the first control socket is
 created.

35. (New) The method of claim 31, wherein (g) further comprises:

2 (5) ceasing polling of the first control socket until completion of execution of
the first control task object.

36. (New) The method of claim 31, wherein (g)(4) comprises:
2 placing the first control task object in a task queue, wherein a set of processor
threads is dedicated to executing task objects placed in said task queue.